

Sys : TP 02 Prise en main Linux

DAKKAR Borhen-eddine

Lycée le Corbusier

BTS SN

September 11, 2020

1 Objectifs du TP

L'objectif de ce tp est de vous familiariser avec l'environnement linux. Il s'agit de vous montrer quelques commandes utiles pour bien maitriser votre système d'exploitation.

2 Découverte du shell

Le mot shell signifie « coquille » en anglais. C'est un outil en mode texte qui permet l'exploitation d'un grand nombre de ressources de l'ordinateur. Le shell représente une autre façon d'interagir avec le système en utilisant un terminal (ou console). Le shell vous indique qu'il est prêt en affichant une invite (ou prompt), en général le caractère dollar (\$) ou supérieur (>), en début de ligne.

On peut distinguer diverses sortes de commandes :

- Des commandes « simples » permettant souvent de manipuler des fichiers sur le disque dur et qui affichent leur résultat directement dans le shell (ls, cp, mv, ...). Ces commandes, puisqu'elles utilisent le shell pour leurs interactions, ne peuvent être exécutées en dehors d'un terminal;
- Des commandes plus sophistiquées qui n'utilisent pas le shell (emacs, gedit, xcalc, ...). Ces programmes peuvent également être exécutés à l'aide d'une icône de l'interface graphique ;

- Des commandes internes au shell (built-in commands) qui permettent d'utiliser certaines facilités offertes par le shell (alias, exit, ...). Il s'agit de commandes internes, servant habituellement à modifier le comportement du shell lui-même.

Les commande simples:

1. **whoami** : Cette commande permet de vous renvoyer votre identifiant.
2. **who** : elle permet de connaître la liste de personnes actuellement connectées sur la machine. Évidemment, pour l'instant vous êtes seul sur votre machine ! À noter que la commande **w** (pour what) est plus complète et vous dit plus précisément qui fait quoi sur la machine.
3. **man** : cette commande vous permet d'accéder au manuel le terminal. Il suffit de taper **man ma_commande** pour accéder à la description complète de **ma_commande**.
4. **cat, less** : La commande **cat mon_fichier** permet d'afficher tout le contenu du fichier texte **mon_fichier**. Pour des textes plus longs, nous utilisons **textbfless** ou **more** de manière à avoir le texte page par page.

3 Qui, quoi (who, what)

3.1

Testez les commandes **who**, **whoami**, **w** et indiquez ce que représentent les champs suivants: IDLE, JCPU,PCPU, WHAT (ou QUOI) et TTY.

3.2

Utilisez la commande **man who** pour afficher le manuel de cette commande. Vous pouvez défiler le texte avec les flèches haut et bas, ou les touches page up et page down. Pour sortir, pressez la touche q. Ensuite donner la ligne de commande permettant d'afficher qui est connecté sur la machine. Affichez la version de cette commande.

Réponse

.....

.....

4 Création d'un répertoire de travail

Nous allons procéder à la création d'un répertoire en ligne de commande nommer TP02_Linux.

L'arborescence de ce répertoire est la suivante: /home/User/Bureau/TP02_Linux avec User le nom d'utilisateur de votre ordinateur.

Allez sur le terminal et écrivez:

\$ mkdir -p /HOME/User/Bureau/TP02_Linux Vérifiez si votre répertoire a été bien créé.

La commande **cd** permet de changer le répertoire courant.

Testez les commandes suivantes et indiquez leurs rôles:

- **ls** :
- **grep** :
- **sort** :
- **touch** :
- **cp** :
- **mv** :
- **rm** :

5 La commande **gedit**

Allez sur le terminal et tapez la commande **gedit** pour lancer un éditeur de texte, puis rédigez le texte suivant :

Je suis un élève de BTS SN

Je suis au lycée le Corbusier

Je suis en 1ère année

Ma spécialité est Informatique et réseaux

Sauvegardez le fichier dans votre répertoire TP02_Linux sous le nom commande_gedit.txt.

6 La commande **cat**

En utilisant la commande **man cat** observez le rôle de **cat**.

Affichez le contenu du fichier `presentation.txt` en utilisant la commande **cat** et :

1. **Le nom absolu du fichier** : chemin absolu identifie une ressource en commençant à la racine de l'arborescence, avec le caractère `/`. **Exemple** (`/home/user/Bureau/TP02_Linux`)
2. **Le nom relatif** : Le chemin relatif identifie une ressource à partir du répertoire courant. **Exemple** (`../TP02_Linux/presentation.txt`).

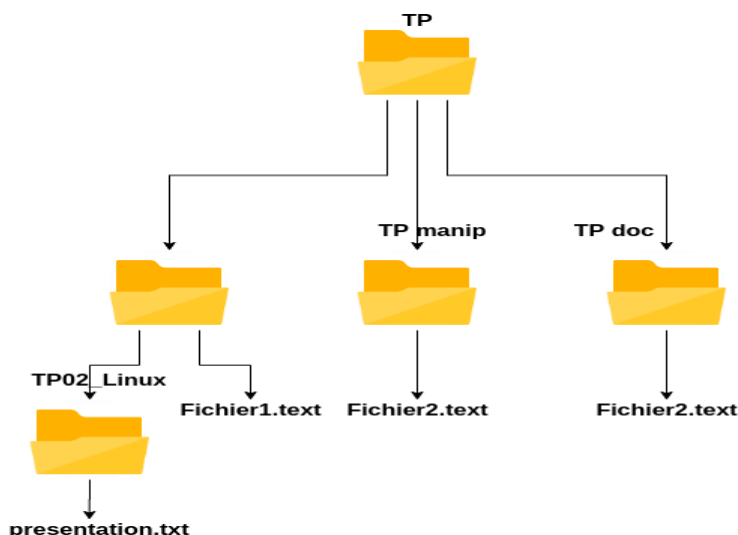
7 Interrompre un programme (le symbole `&`)

1. Ouvrez le fichier texte "`presentation.txt`" avec l'éditeur de texte en tapant la commande `gedit` dans votre terminal.
2. Revenez sur le terminal et tapez une autre commande, par exemple `who`. Que se passe-t-il?
.....
3. Répétez les étapes 1 et 2, en ajoutant cette fois-ci le caractère `&` en fin de commande, `gedit presentation.txt &`. Indiquez la différence.
.....

8 Arborescence

À partir de votre répertoire Système reproduisez l'arborescence de la Figure 1.

Figure 1: Exemple d'arborescence



Vérifiez si vous n'avez rien oublié en utilisant la commande **ls -R** depuis votre répertoire personnel. Pouvez-vous expliquer son rôle ?

.....

9 Copie et suppression de fichiers

1. Créez une copie de "presentation.txt" appelée **.plan** (le point est voulu) dans le répertoire TP02_Linux.
2. Supprimez ensuite le fichier "presentation.txt" à l'aide de la commande **rm**.
3. Affichez le contenu de votre répertoire personnel avec **ls**.

4. Que constatez-vous ?

.....
Réessayez avec la commande `ls -a`. Cette commande permet d'afficher les fichiers cachés(l'argument -a signifie «all» en anglais).

5. Utilisez la commande `cp -r` pour faire une copie du répertoire TP01_Linux dans le répertoire TP manip en lui donnant le nom suivant "Copie_TP02_Linux".
6. À partir de votre répertoire maison (home), essayez de détruire le répertoire **TP doc** puis le répertoire **Copie_TP02_Linux** à l'aide de la commande `rmdir`.
Que se passe-t-il?

.....
Trouvez une solution en utilisant la commande `rm` avec la bonne option (utilisez la commande `man` pour voir les différentes options).

7. Testez la commande `mv` en déplaçant des fichiers dans votre arborescence. Essayez également de déplacer un répertoire. Que doit-on faire pour que cela fonctionne?

.....

10 L'historique des commandes

La commande `history` affiche la liste des commandes récentes, chacune précédée d'un numéro. En tapant `!n` où n est un numéro dans l'historique, on rappelle la commande correspondante.

11 Commandes et options

Cette partie vise à vous habituer à lecture du manuel des commandes. Une bonne pratique c'est de trouver par vous-même les informations dont vous avez besoin.

11.1 La commande ls

Par défaut, le retour de la commande `ls` est un peu aride. Une option permet de colorer l'affichage pour repérer plus rapidement le type des fichiers présents

dans un répertoire. Quelle est cette option?

.....

11.2 La commande `rm`

Par défaut, la commande `rm` ne demande pas de confirmation lorsque vous voulez supprimer un fichier. Ceci peut se révéler dangereux en ce qui concerne la perte d'informatio. Trouvez l'option qui permet de demandez confirmation?

.....

11.3 Les Alias

Les alias sont des substitutions abrégées de commandes répétitives et/ou longues à taper dans la console [1].

- **Syntaxe :** `alias nom_de_votre_alias = 'commande de votre alias'`
- Lorsque vous saisirez `nom_de_votre_alias` dans un terminal, ce sera réellement la commande de votre alias qui sera exécutée.
- Pour retirer un alias, il suffit de saisir la commande `unalias`.

11.3.1 Mon premier alias

Créez une commande «lister» qui liste tous les fichiers d'un répertoire (y compris les fichiers cachés).

11.3.2 Modification d'une commande

Modifier le fonctionnement de la commande `rm` pour qu'elle demande systématiquement la confirmation avant d'effectuer une suppression. Changez en suite le fonctionnement de la commande `ls` afin d'utiliser systématiquement l'affichage coloré.

Ouvrez un nouveau terminal et testez les alias créés. Que constatez vous?

.....

11.4 Fichiers de configuration

Comme vous avez pu le constater, les alias sont effectifs que dans le terminal ouvert.

Pour cela, il est possible de définir les alias dans deux fichiers cachés qui se trouvent dans votre Dossier Personnel:

Dans le fichier `.bashrc` juste après la ligne "some more ls aliases"

Ou bien dans un fichier `.bash_aliases`. Ce sont des fichiers texte contenant des commandes qui sont lues au moment de l'ouverture d'un terminal. Tout changement dans ces fichiers ne sera donc pris en compte que si on relance le shell.

11.4.1 Editer le fichier `.bash_aliases`

Observez le fichier `.bashrc` en tapant la commande `gedit ~/.bashrc`.

Maintenant éditer le fichier `.bash_aliases` (s'il n'existe pas créez-le). Rajouter les alias créés précédemment. Pensez à ouvrir un nouveau terminal afin de rendre les changements effectifs ou à utiliser la commande `source ~/.bash_aliases`.

11.5 Informations sur les utilisateurs

Les commandes `users`, `groups` et `id` vous aident à savoir plus d'informations sur les utilisateurs connectés avec le user. Testez ces commandes et identifiez les groupes auxquels vous appartenez et donnez les numéros de ces groupes.

.....

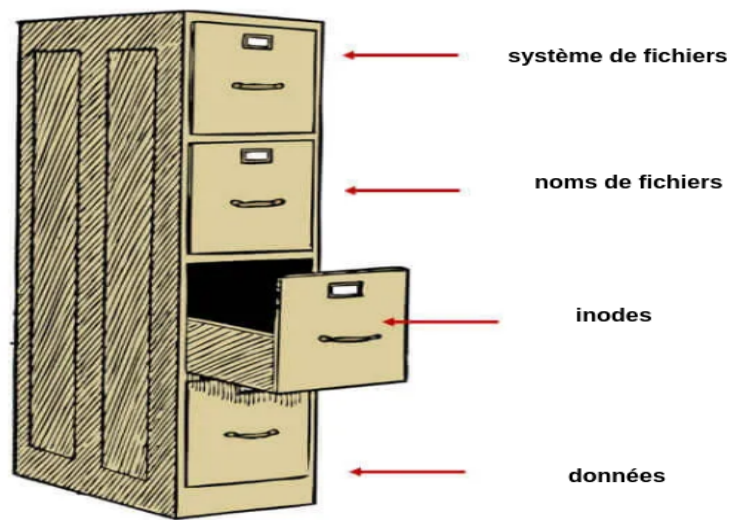
12 Les fichiers sous Linux

Un système de fichiers est divisé en deux parties: les blocs de données et les inodes. Le nombre de blocs est fixe une fois créé et ne peut pas être modifié. Le nom, le chemin, l'emplacement, les liens et autres attributs de fichier ne se trouvent pas dans le répertoire. Les répertoires sont simplement des tables contenant les noms des fichiers avec le numéro d'inode correspondant.

La partie supérieure de la hiérarchie est le système de fichiers lui-même. Dans le système de fichiers se trouvent les noms de fichiers. Les noms de fichiers

sont liés aux inodes. Les inodes sont liés aux données. La Figure 2 présente cette architecture.

Figure 2: L'architecture des fichiers Linux



La commande `ls -l` permet de connaître toutes les caractéristiques des fichiers, y compris leur nombre de liens.

12.1 Liens entre les fichiers

12.1.1 Lien physique

Un lien physique sous Linux est une entrée de répertoire en double. Les deux entrées de répertoire pointent vers le même fichier (ou plus précisément, vers le même inode). Aucune des deux entrées n'a de priorité sur l'autre et les deux sont également valides. Pour supprimer le fichier, vous devez supprimer les deux liens physiques vers le fichier.

Pour créer un lien physique, vous pouvez utiliser la commande `ln` sans aucune option. Nous allons créer un lien physique pour notre fichier `presentation.txt` comme suit:

- `ln presentation.txt lien_physique.lnk`

12.1.2 Lien symbolique

Un lien symbolique sous Linux est un type spécial de fichier qui pointe vers d'autres fichiers, au lieu de pointer vers des données sur le disque dur. Contrairement aux liens physiques, les liens symboliques ne partagent pas le même numéro d'inode. Un lien symbolique contient une chaîne qui est automatiquement interprétée et suivie par le système d'exploitation comme un chemin vers un autre fichier ou répertoire. Un lien symbolique est un deuxième fichier qui existe indépendamment de sa cible.

Pour créer un lien symbolique, utilisez l'option -s avec la commande `ln`, comme illustré dans l'exemple ci-dessous:

- `ln -s Nouveau document lien_symb.lnk`

Quelles est la différences entre les liens physiques et les liens symboliques?

.....

13 La commande `chmod` (change file mode bits)

`chmod` liste-droit fichier permet d'accorder ou de retirer au(x) fichier(s) selon le mode, qui peut être soit une représentation symbolique des modifications à apporter, soit un nombre octal représentant le modèle de bits pour les nouveaux bits de mode. Une liste est de la forme **droit1,...,droitn** où chaque droit i peut être par exemple :

- **u+r** pour rajouter au propriétaire le droit en lecture.
- **g-w** pour retirer aux membres du groupe le droit en écriture.
- **o+x** pour donner aux autres utilisateurs le droit en exécution.

13.1 Effet de `chmod` sur les liens

Pour voir l'effet de `chmod` sur les liens physiques et symboliques, créez deux fichiers "test1.txt" et "test2.txt" dans le répertoire "TP manip". Puis créez un lien physique pour test1 et un lien symbolique pour text2 (text1_phys.lnk, test2_symb.lnk).

1. A l'aide de `ls -l` notez les permissions de chaque fichier avec ces liens.

.....

2. Utilisez `chmod` pour modifier les droits de `test1.txt` (`chmod 777`). Que se passe-t-il pour `"text1_phys"`?

.....

3. Modifiez maintenant les droits de `"text1_phys"` en utilisant la commande `chmod 665`, que se passe-t-il pour `"test1.txt"` ?

4. Faites de même pour le lien symbolique.

.....

.....

Remarque : la commande `echo "mon texte" > nom_fichier`, vous permet d'écrire le texte «mon texte» dans le fichier `mon_fichier`.

13.2 Essai de `chmod`

1. Dans le répertoire `"TP doc"`, créez un fichier `"essai.txt"`, et écrivez la phrase `"comment fonctionne chmod"`.
2. Notez les permissions actuelles du répertoire `"TP doc"` et du fichier `"essai.txt"`.
3. En utilisant la commande `chmod`, retirez-vous le droit en lecture et en écriture sur le fichier `"essai.txt"`. Vérifiez l'effet obtenu en essayant d'afficher le contenu du fichier sur la fenêtre du terminal, puis de remplacer ce contenu par une phrase différente.

13.3 Effet `chmod` sur les fichiers et les répertoires

1. Placez-vous dans le répertoire `"TP doc"`, et retirez-vous le droit en lecture pour ce répertoire. Listez le contenu du répertoire avec `ls`, puis affichez le contenu du fichier `"essai.txt"`. Qu'en déduisez-vous?

.....

Rétablissez le droit en lecture sur le répertoire `"TPdoc"`.

2. Créez un fichier `"Test3"` dans `"TPdoc"`:

- Retirez au fichier "Test3" et au répertoire "TPdoc" le droit en écriture. Essayez de modifier le fichier "Test3" puis le supprimer. Que constatez-vous?

.....

- Rétablissez ensuite le droit en écriture au fichier "Test3", supprimez le droit en écriture au répertoire "TP doc". Essayez de modifier le fichier "Test3", puis de le supprimer. Que constatez-vous?

.....

•

13.4 **chmod** en octal

Les commandes chmod et umask permettent aussi d'utiliser une syntaxe de la forme chmod nnn, où chaque n est un nombre entre 0 et 7 (notation octale).

Valeur	Description
0	- - - (aucun droit)
1	- - x (exécution)
2	- w - (écriture)
3	- w x (écriture et exécution)
4	r - - (lecture seule)
5	r - x (lecture et exécution)
6	r w - (lecture et écriture)
7	r w x (lecture, écriture et exécution)

Exemple :

Pour rwx, on aura : $4+2+1 = 7$

Pour rw-, on aura : $4+2+0 = 6$

Pour r-, on aura : $4+0+0 = 4$

- Visualisez les droits des répertoires et fichiers inclus dans le répertoire «TP». Maintenant modifiez le mask en supprimant le droit de lecture aux utilisateurs de son propre groupe. essayez de faire une lecture? Notez ce que vous observez?

.....

References

- [1] *alias*. https://doc.ubuntu-fr.org/aliasajouter_un_alias_provisoirement. 2020.